Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
L5	182	(shar$4 near4 (cach$3 memory storage directory)) same ((perman$4 persist$4 delicat$4 maintain$4) near6 (ownership own$4))	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/02/18 17:44
L6	148	state and L5	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/02/18 17:44
L7	108	((multiple mult$5) near4 (process$4 computer cpu host node cluster)) and 6	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/02/18 17:44
L8	969	711/141.ccls.	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/02/18 17:45
L9	18	7 and 8	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/02/18 17:45
S1	438	(shar$4 near4 (cach$3 memory storage directory)) same ((perman$4 persist$4 delicat$4 exclus$5) near6 (ownership own$4))	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/02/18 14:40
S2	381	((multiple mult$5) near4 (process$4 computer cpu host node cluster)) and S1	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/02/18 14:40
S3	238	state same S1	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/02/18 14:41
S4	207	((multiple mult$5) near4 (process$4 computer cpu host node cluster)) and S3	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/02/18 14:41
S5	565	(shar$4 near4 (cach$3 memory storage directory)) same ((perman$4 persist$4 delicat$4 exclus$5 maintain$4) near6 (ownership own$4))	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/02/18 15:39
S6	272	state same S5	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/02/18 15:40
S7	228	((multiple mult$5) near4 (process$4 computer cpu host node cluster)) and S6	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/02/18 17:44

S8	969	711/141.ccls.	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/02/18 17:45
S9	80	S7 and S8	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/02/18 14:42
S10	182	(shar$4 near4 (cach$3 memory storage directory)) same ((perman$4 persist$4 delicat$4 maintain$4) near6 (ownership own$4))	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/02/18 15:40
S11	48	state same S10	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/02/18 17:44
S12	14	S1 and S11	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/02/18 15:40

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
L1	7039	((multiple mult$5) near4 (process$3 computer cpu host node cluster)) same (shar$4 with (memory storage cach$3 device))	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/02/17 17:59
L2	82920	(cach$4 storage memory) with ((line entry data) with (ownership state))	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/02/17 18:02
L3	546	((only exculsiv$3) with (second$3 near4 (cach$3 memory storage))) same 2	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/02/17 18:04
L4	35	(without with (invalidat$4 modif$4) with (entry line data)) and 1 and 3	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/02/17 18:07
L5	32	((stor$3 cop$3 duplica$4 writ$3) near3 back) and 4	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/02/17 18:20
L6	1327	((state ownership) with (entry line block data)) with (in stay only exculs$5) with (second$3 near4 (cach$3 memory storage))	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/02/17 18:22
L7	1545	((state ownership) with (entry line block data)) with (in stay only exculs$5 maintain$4) with (second$3 near4 (cach$3 memory storage))	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/02/17 18:24
L8	110	1 and 7	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/02/17 19:07

acm PORTAL
US Patent & Trademark Office

Subscribe (Full Service) Register (Limited Service, Free) Login

Search: ● The ACM Digital Library ○ The Guide

+ownership +share +cache +memory +coherency +state +m SEARCH

THE ACM DIGITAL LIBRARY

Feedback Report a problem Satisfaction s

Terms used
ownership share cache memory coherency state maintaining multiple processor permant persisting maint

Sort results by relevance
Display results expanded form

Save results to a Binder
Search Tips
□ Open results in a new window

Try an Advanced Search
Try this search in The ACM Gui

Results 1 - 20 of 83 Result page: **1** 2 3 4 5 next

Relevance scale

1 Sharing and protection in a single-address-space operating system
Jeffrey S. Chase, Henry M. Levy, Michael J. Feeley, Edward D. Lazowska
November 1994 **ACM Transactions on Computer Systems (TOCS)**, Volume 12 Issue 4
Full text available: pdf(2.87 MB) Additional Information: full citation, abstract, references, citings, index terms

This article explores memory sharing and protection support in Opal, a single-address-space opera system designed for wide-address (64-bit) architectures. Opal threads execute within protection domains in a single shared virtual address space. Sharing is simplified, because addresses are con independent. There is no loss of protection, because addressability and access are independent; th right to access a segment is determined by the protection domain in which a thread executes. T ..

Keywords: 64-bit architectures, capability-based systems, microkernel operating systems, object oriented database systems, persistent storage, protection, single-address-space operating system wide-address architectures

2 Synchronization with multiprocessor caches
Joonwon Lee, Umakishore Ramachandran
May 1990 **ACM SIGARCH Computer Architecture News , Proceedings of the 17th annual international symposium on Computer Architecture**, Volume 18 Issue 3
Full text available: pdf(1.18 MB) Additional Information: full citation, abstract, references, citings, index terms

Introducing private caches in bus-based shared memory multiprocessors leads to the cache consis problem since there may be multiple copies of shared data. However, the ability to snoop on the b coupled with the fast broadcast capability allows the design of special hardware support for synchronization. We present a new lock-based cache scheme which incorporates synchronization i the cache coherency mechanism. With this scheme high-level synchronization primitives as well as le ...

3 Using prediction to accelerate coherence protocols
Shubhendu S. Mukherjee, Mark D. Hill
April 1998 **ACM SIGARCH Computer Architecture News , Proceedings of the 25th annual international symposium on Computer architecture**, Volume 26 Issue 3
Full text available: pdf(1.71 MB) Publisher Site Additional Information: full citation, abstract, references, citings, index terms

Most large shared-memory multiprocessors use directory protocols to keep per-processor caches coherent. Some memory references in such systems, however, suffer long latencies for misses to remotely-cached blocks. To ameliorate this latency, researchers have augmented standard cohere protocols with optimizations for specific sharing patterns, such as read-modify-write, producer-consumer, and migratory sharing. This paper seeks to replace these directed solutions with generi prediction logic t ...

4 C^{2}MP: a cache-coherent, distributed memory multiprocessor-system
D. E. Marquardt, H. S. Alkhatib
August 1989 **Proceedings of the 1989 ACM/IEEE conference on Supercomputing**
Full text available: pdf(1.22 MB) Additional Information: full citation, abstract, references, citings, index terms

Current research into the problems of cache coherency in multiprocessor (MP) systems, has prima focused on bus based memory interconnection networks (M-ICN) and the use of various types of "snooping" cache coherency protocols. Bus bandwidth limitations can be alleviated through the us wider bandwidth general interconnection structures, such as a crossbar switch. However, if private caches are used, the cache coherency problem becomes mul ...

5 A characterization of sharing in parallel programs and its application to coherency protocol evaluation
S. J. Eggers, R. H. Katz
May 1988 **ACM SIGARCH Computer Architecture News , Proceedings of the 15th Annual International Symposium on Computer architecture**, Volume 16 Issue 2
Full text available: pdf(1.38 MB) Additional Information: full citation, abstract, references, citings, index terms

In this paper we use trace-driven simulation to analyze the memory reference patterns of write sh data in several parallel applications. We first develop a characterization of write sharing (based on notion of a write run), and then examine the traces, using metrics derived from the characterizati The results indicate that the amount of write sharing in all programs is small; and that it is charac by short to medium sequences of per processor references, with little conten ...

6 Techniques for reducing consistency-related communication in distributed shared-memory systems
John B. Carter, John K. Bennett, Willy Zwaenepoel
August 1995 **ACM Transactions on Computer Systems (TOCS)**, Volume 13 Issue 3
Full text available: pdf(2.86 MB) Additional Information: full citation, abstract, references, citings, index terms,

Distributed shared memory (DSM) is an abstraction of shared memory on a distributed-memory machine. Hardware DSM systems support this abstraction at the architecture level; software DSM systems support the abstraction within the runtime system. One of the key problems in building a efficient software DSM system is to reduce the amount of communication needed to keep the distr memories consistent. In this article we present four techniques for doing so: software release consistency; m ...

Keywords: cache consistency protocols, distributed shared memory, memory models, release consistency, virtual shared memory

7 Cache memory performance in a unix enviroment
Cedell Alexander, William Keshlear, Furrokh Cooper, Faye Briggs
June 1986 **ACM SIGARCH Computer Architecture News**, Volume 14 Issue 3
Full text available: pdf(2.10 MB) Additional Information: full citation, citings, index terms

8 CACHET: an adaptive cache coherence protocol for distributed shared-memory systems
Xiaowei Shen, Arvind, Larry Rudolph

May 1999 **Proceedings of the 13th international conference on Supercomputing**

Full text available: pdf(1.34 MB) Additional Information: full citation, references, citings, index terms

9 Evaluating the performance of four snooping cache coherency protocols

S. J. Eggers, R. H. Katz

April 1989 **ACM SIGARCH Computer Architecture News , Proceedings of the 16th annual international symposium on Computer architecture**, Volume 17 Issue 3

Full text available: pdf(1.70 MB) Additional Information: full citation, abstract, references, citings, index terms

Write-invalidate and write-broadcast coherency protocols have been criticized for being unable to achieve good bus performance across all cache configurations. In particular, write-invalidate performance can suffer as block size increases; and large cache sizes will hurt write-broadcast. Re broadcast and competitive snooping extensions to the protocols have been proposed to solve each problem. Our results indicate that the benefits of the extensions are limited. Read-broadcast ...

10 Implementing a cache consistency protocol

R. H. Katz, S. J. Eggers, D. A. Wood, C. L. Perkins, R. G. Sheldon

June 1985 **ACM SIGARCH Computer Architecture News , Proceedings of the 12th annual international symposium on Computer architecture**, Volume 13 Issue 3

Full text available: pdf(803.11 KB) Additional Information: full citation, citings, index terms

Keywords: ownership-based protocols, shared bus multicomprocessor cache consistency, single implementation, snooping caches

11 Correct memory operation of cache-based multiprocessors

C. Scheurich, M. Dubois

June 1987 **Proceedings of the 14th annual international symposium on Computer architect**

Full text available: pdf(1.05 MB) Additional Information: full citation, abstract, references, citings, index terms

This paper shows that cache coherence protocols can implement indivisible synchronization primiti reliably and can also enforce sequential consistency. Sequential consistency provides a commonly accepted model of behavior of multiprocessors. We derive a simple set of conditions needed to enf sequential consistency in multiprocessors. These conditions are easily applied to prove the correct existing cache coherence protocols that rely on one or multiple broadcast buses to enfor ...

12 The effect of sharing on the cache and bus performance of parallel programs

S. J. Eggers, R. H. Katz

April 1989 **ACM SIGARCH Computer Architecture News , Proceedings of the third internatio conference on Architectural support for programming languages and operating systems**, Volume 17 Issue 2

Full text available: pdf(1.62 MB) Additional Information: full citation, abstract, references, citings, index terms,

Bus bandwidth ultimately limits the performance, and therefore the scale, of bus-based, shared m multiprocessors. Previous studies have extrapolated from uniprocessor measurements and simulat to estimate the performance of these machines. In this study, we use traces of parallel programs t evaluate the cache and bus performance of shared memory multiprocessors, in which coherency is maintained by a write-invalidate protocol. In particular, we analyze the effect of sharing ...

13 A cache coherence approach for large multiprocessor systems

J. K. Archibald

June 1988 **Proceedings of the 2nd international conference on Supercomputing**

Full text available: pdf(1.05 MB) Additional Information: full citation, abstract, references, citings, index terms

This paper explores the architecture of high-performance large scale multiprocessors using private caches for each processor. The caches reduce the average memory access time, but they also resi the well known cache coherence problem. Multiple copies of each memory location are allowed to · but they must be kept consistent with each other. In this paper, we present a solution to the cach coherence problem specifically for shared bus multiprocessors that adapts dyn ...

14 Transactional client-server cache consistency: alternatives and performance
Michael J. Franklin, Michael J. Carey, Miron Livny
September 1997 **ACM Transactions on Database Systems (TODS)**, Volume 22 Issue 3

Full text available: pdf(452.41 KB) Additional Information: full citation, abstract, references, citings, index terms,

Client-server database systems based on a data shipping model can exploit client memory resourc caching copies of data items across transaction boundaries. Caching reduces the need to obtain dá from servers or other sites on the network. In order to ensure that such caching does not result in violation of transaction semantics, a transactional cache consistency maintenance algorithm is req Many such algorithms have been proposed in the literature and, as all provide the sam ...

15 An evaluation of directory schemes for cache coherence
Anant Agarwal, Richard Simoni, John Hennessy, Mark Horowitz
August 1998 **25 years of the international symposia on Computer architecture (selected pap**

Full text available: pdf(1.31 MB) Additional Information: full citation, references, index terms

16 An evaluation of directory schemes for cache coherence
A. Agarwal, R. Simoni, J. Hennessy, M. Horowitz
May 1988 **ACM SIGARCH Computer Architecture News , Proceedings of the 15th Annual International Symposium on Computer architecture**, Volume 16 Issue 2

Full text available: pdf(1.35 MB) Additional Information: full citation, abstract, references, citings, index terms

The problem of cache coherence in shared-memory multiprocessors has been addressed using twc approaches: directory schemes and snoopy cache schemes. Directory schemes have been given le attention in the past several years, while snoopy cache methods have become extremely popular. Directory schemes for cache coherence are potentially attractive in large multiprocessor systems t are beyond the scaling limits of the snoopy cache schemes. Slight modifications to directory schen can ...

17 Design and performance of the Shasta distributed shared memory protocol
Daniel J. Scales, Kourosh Gharachorloo
July 1997 **Proceedings of the 11th international conference on Supercomputing**

Full text available: pdf(1.40 MB) Additional Information: full citation, references, citings, index terms

18 SoftFLASH: analyzing the performance of clustered distributed virtual shared memory
Andrew Erlichson, Neal Nuckolls, Greg Chesson, John Hennessy
September 1996 **Proceedings of the seventh international conference on Architectural suppo programming languages and operating systems**, Volume 31 , 30 Issue 9 , 5

Full text available: pdf(1.29 MB) Additional Information: full citation, abstract, references, citings, index terms

One potentially attractive way to build large-scale shared-memory machines is to use small-scale · medium-scale shared-memory machines as clusters that are interconnected with an off-the-shelf network. To create a shared-memory programming environment across the clusters, it is possible a virtual shared-memory software layer. Because of the low latency and high bandwidth of the interconnect available within each cluster, there are clear advantages in making the clusters as lar possi ...

19 The VMP multiprocessor: initial experience, refinements, and performance evaluation
D. R. Cheriton, A. Gupta, P. D. Boyle, H. A. Goosen
May 1988 **ACM SIGARCH Computer Architecture News , Proceedings of the 15th Annual International Symposium on Computer architecture**, Volume 16 Issue 2
Full text available: pdf(1.73 MB) Additional Information: full citation, abstract, references, citings, index terms

VMP is an experimental multiprocessor being developed at Stanford University, suitable for high-performance workstations and server machines. Its primary novelty lies in the use of software management of the per-processor caches and the design decisions in the cache and bus that mak approach feasible. The design and some uniprocessor trace-driven simulations indicating its perfor have been reported previously. In this paper, we present our initial experience with the V ...

20 Using dataflow analysis techniques to reduce ownership overhead in cache coherence protc
Jonas Skeppstedt, Per Stenström
November 1996 **ACM Transactions on Programming Languages and Systems (TOPLAS)**, Volume Issue 6
Full text available: pdf(284.68 KB) Additional Information: full citation, abstract, references, index terms, review

In this article, we explore the potential of classical dataflow analysis techniques in removing overh write-invalidate cache coherence protocols for shared-memory multiprocessors. We construct the compiler algorithms with varying degree of sophistication that detect loads followed by stores to th same address. Such loads are marked and constitute a hint to the cache to obtain an exclusive co the block so that the subsequent store does not introduce access penalties. The simplest ...

Keywords: cache coherence, dataflow analysis, performance evaluation

Results 1 - 20 of 83 Result page: **1** 2 3 4 5 next

PORTAL

US Patent & Trademark Office

Subscribe (Full Service) Register (Limited Service, Free) Login

Search: ● The ACM Digital Library ○ The Guide

+ownership +share +cache +memory +coherency +state +m SEARCH

THE ACM DIGITAL LIBRARY

Feedback Report a problem Satisfaction s

Terms used
ownership share cache memory coherency state maintaining multiple processor permant persisting maint

Sort results by relevance

Display results expanded form

Save results to a Binder

Search Tips

□ Open results in a new window

Try an Advanced Search
Try this search in The ACM Gui

Results 21 - 40 of 83 Result page: previous 1 **2** 3 4 5 next

Relevance scale

21 Memory coherence in shared virtual memory systems
Kai Li, Paul Hudak
November 1989 **ACM Transactions on Computer Systems (TOCS)**, Volume 7 Issue 4
Full text available: pdf(2.71 MB) Additional Information: full citation, abstract, references, citings, index terms,

The memory coherence problem in designing and implementing a shared virtual memory on loose coupled multiprocessors is studied in depth. Two classes of algorithms, centralized and distributed solving the problem are presented. A prototype shared virtual memory on an Apollo ring based on algorithms has been implemented. Both theoretical and practical results show that the memory coherence problem can indeed be solved efficiently on a loosely coupled multiprocessor.

22 VM-based shared memory on low-latency, remote-memory-access networks
Leonidas Kontothanassis, Galen Hunt, Robert Stets, Nikolaos Hardavellas, Michał Cierniak, Srinivasar Parthasarathy, Wagner Meira, Sandhya Dwarkadas, Michael Scott
May 1997 **ACM SIGARCH Computer Architecture News , Proceedings of the 24th annual international symposium on Computer architecture**, Volume 25 Issue 2
Full text available: pdf(1.96 MB) Additional Information: full citation, abstract, references, citings, index terms

Recent technological advances have produced network interfaces that provide users with very low latency access to the memory of remote machines. We examine the impact of such networks on tl implementation and performance of software DSM. Specifically, we compare two DSM systems--- Cashmere and TreadMarks---on a 32-processor DEC Alpha cluster connected by a Memory Channe network.Both Cashmere and TreadMarks use virtual memory to maintain coherence on pages, anc use lazy, multi-writer releas ...

23 Empirical performance evaluation of concurrency and coherency control protocols for databa sharing systems
Erhard Rahm
June 1993 **ACM Transactions on Database Systems (TODS)**, Volume 18 Issue 2
Full text available: pdf(3.37 MB) Additional Information: full citation, abstract, references, citings, index terms,

Database Sharing (DB-sharing) refers to a general approach for building a distributed high perforn transaction system. The nodes of a DB-sharing system are locally coupled via a high-speed intercc and share a common database at the disk level. This is also known as a "shared disk" approach. W compare database sharing with the database partitioning (shared nothing) approach and discuss t functional DBMS components that require new and coordinated solutions for DB-shar ...

Keywords: coherency control, concurrency control, database partitioning, database sharing, performance analysis, shared disk, shared nothing, trace-driven simulation

24 Adjustable block size coherent caches
Czarek Dubnicki, Thomas J. LeBlanc
April 1992 **ACM SIGARCH Computer Architecture News , Proceedings of the 19th annual international symposium on Computer architecture**, Volume 20 Issue 2
Full text available: pdf(1.24 MB) Additional Information: full citation, abstract, references, citings, index terms

Several studies have shown that the performance of coherent caches depends on the relationship between the granularity of sharing and locality exhibited by the program and the cache block size. cache blocks exploit processor and spatial locality, but may cause unnecessary cache invalidations to false sharing. Small cache blocks can reduce the number of cache invalidations, but increase th nuber of bus or network transactions required to load data into the cache. In this paper we ...

25 Piranha: a scalable architecture based on single-chip multiprocessing
Luiz André Barroso, Kourosh Gharachorloo, Robert McNamara, Andreas Nowatzyk, Shaz Qadeer, Bar Sano, Scott Smith, Robert Stets, Ben Verghese
May 2000 **ACM SIGARCH Computer Architecture News , Proceedings of the 27th annual international symposium on Computer architecture**, Volume 28 Issue 2
Full text available: pdf(191.10 KB) Additional Information: full citation, abstract, references, citings, index terms

The microprocessor industry is currently struggling with higher development costs and longer desi times that arise from exceedingly complex processors that are pushing the limits of instruction-lev parallelism. Meanwhile, such designs are especially ill suited for important commercial applications as on-line transaction processing (OLTP), which suffer from large memory stall times and exhibit li instruction-level parallelism. Given that commercial applications constitute by fa ...

26 Adaptive, fine-grained sharing in a client-server OODBMS: a callback-based approach
Markos Zaharioudakis, Michael J. Carey, Michael J. Franklin
December 1997 **ACM Transactions on Database Systems (TODS)**, Volume 22 Issue 4
Full text available: pdf(441.80 KB) Additional Information: full citation, abstract, references, citings, index terms,

For reasons of simplicity and communication efficiency, a number of existing object-oriented datat management systems are based on page server architectures; data pages are their minimum unit transfer and client caching. Despite their efficiency, page servers are often criticized as being too retrictive when it comes to concurrency, as existing systems use pages as the minimum locking ur well. In this paper we show how to support object-level locking in a page-server context. Sev ...

Keywords: cache coherency, cache consistency, client-server databased, fine-grained sharing, ot oriented databases, performance analysis

27 Performance of database workloads on shared-memory systems with out-of-order processoı
Parthasarathy Ranganathan, Kourosh Gharachorloo, Sarita V. Adve, Luiz André Barroso
October 1998 **Proceedings of the eighth international conference on Architectural support fo programming languages and operating systems**, Volume 33 , 32 Issue 11 , 5
Full text available: pdf(1.62 MB) Additional Information: full citation, abstract, references, citings, index terms

Database applications such as online transaction processing (OLTP) and decision support systems constitute the largest and fastest-growing segment of the market for multiprocessor servers. How most current system designs have been optimized to perform well on scientific and engineering workloads. Given the radically different behavior of database workloads (especially OLTP), it is imţ to re-evaluate key system design decisions in the context of this important class of applicatio ...

28
Multi-level shared caching techniques for scalability in VMP-M/C

D. R. Cheriton, H. A. Goosen, P. D. Boyle
April 1989 **ACM SIGARCH Computer Architecture News , Proceedings of the 16th annual international symposium on Computer architecture**, Volume 17 Issue 3
Full text available: pdf(1.27 MB) Additional Information: full citation, abstract, references, citings, index terms

The problem of building a scalable shared memory multiprocessor can be reduced to that of buildi scalable memory hierarchy, assuming interprocessor communication is handled by the memory sy In this paper, we describe the VMP-MC design, a distributed parallel multi-computer based on the multiprocessor design, that is intended to provide a set of building blocks for configuring machine one to several thousand processors. VMP-MC uses a memory hierarchy based on shared caches ...

29 A class of compatible cache consistency protocols and their support by the IEEE futurebus
P. Sweazey, A. J. Smith
June 1986 **ACM SIGARCH Computer Architecture News , Proceedings of the 13th annual international symposium on Computer architecture**, Volume 14 Issue 2
Full text available: pdf(1.05 MB) Additional Information: full citation, abstract, references, citings, index terms

Standardization of a high performance blackplane bus, so that it can accommodate boards develo different vendors, implies the need for a standardized cache consistency protocol. In this paper w define a class of compatible consistency protocols supported by the current IEEE Futurebus design refer to this class as the MOESI class of protocols; the term "MOESI" is derived from the names of states. This class of protocols has the property that any system component ca ...

30 An interaction of coherence protocols and memory consistency models in DSM systems
Weisong Shi, Weiwu Hu, Zhimin Tang
October 1997 **ACM SIGOPS Operating Systems Review**, Volume 31 Issue 4
Full text available: pdf(1.09 MB) Additional Information: full citation, abstract, citings, index terms

Coherence protocols and memory consistency models are two improtant issues in hardware coher shared memory multiprocessors and softare distributed shared memory(DSM) systems. Over the many researchers have made extensive study on these two issues repectively. However, the inter between them has not been studied in the literature. In this paper, we study the coherence protoc and memory consistency models used by hardware and software DSM systems in detail. Based on analysis ...

Keywords: coherence protocol, event ordering, hardware DSM systems, memory consistency mo software DSM systems

31 Efficient strategies for software-only protocols in shared-memory multiprocessors
Håkan Grahn, Per Stenström
May 1995 **ACM SIGARCH Computer Architecture News , Proceedings of the 22nd annual international symposium on Computer architecture**, Volume 23 Issue 2
Full text available: pdf(1.31 MB) Additional Information: full citation, abstract, references, citings, index terms

The cost, complexity, and inflexibility of hardware-based directory protocols motivate us to study performance implications of protocols that emulate directory management using software handler executed on the compute processors. An important performance limitation of such software-only protocols is that software latency associated with directory management ends up on the critical m access path for read miss transactions. We propose five strategies that support efficient data transfers ...

32 Cache coherence protocols: evaluation using a multiprocessor simulation model
James Archibald, Jean-Loup Baer
September 1986 **ACM Transactions on Computer Systems (TOCS)**, Volume 4 Issue 4
Full text available: pdf(1.79 MB) Additional Information: full citation, abstract, references, citings, index terms,

Using simulation, we examine the efficiency of several distributed, hardware-based solutions to th cache coherence problem in shared-bus multiprocessors. For each of the approaches, the associat protocol is outlined. The simulation model is described, and results from that model are presented magnitude of the potential performance difference between the various approaches indicates that choice of coherence solution is very important in the design of an efficient shared-bus multi ...

33 Munin: distributed shared memory based on type-specific memory coherence
J. K. Bennett, J. B. Carter, W. Zwaenepoel
February 1990 **ACM SIGPLAN Notices , Proceedings of the second ACM SIGPLAN symposium Principles & practice of parallel programming**, Volume 25 Issue 3
Full text available: pdf(1.05 MB) Additional Information: full citation, abstract, references, citings, index terms

We are developing Munin, a system that allows programs written for shared memory multiprocess be executed efficiently on distributed memory machines. Munin attempts to overcome the architec limitations of shared memory machines, while maintaining their advantages in terms of ease of programming. Our system is unique in its use of loosely coherent memory, based on the partial or specified by a shared memory parallel program, and in its use of type-specific memory coherence.

34 Simple compiler algorithms to reduce ownership overhead in cache coherence protocols
Jonas Skeppstedt, Per Stenström
November 1994 **Proceedings of the sixth international conference on Architectural support fc programming languages and operating systems**, Volume 29 , 28 Issue 11 , 5
Full text available: pdf(1.47 MB) Additional Information: full citation, abstract, references, citings, index terms

We study in this paper the design and efficiency of compiler algorithms that remove ownership ov in shared-memory multiprocessors with write-invalidate protocols. These algorithms detect loads followed by stores to the same address. Such loads are marked and constitute a hint to the cache obtain an exclusive copy of the block. We consider three algorithms where the first one focuses or store sequences within each basic block of code and the other two analyse the existence of l ...

35 Multiple vs. wide shared bus multiprocessors
A. Hopper, A. Jones, D. Lioupis
April 1989 **ACM SIGARCH Computer Architecture News , Proceedings of the 16th annual international symposium on Computer architecture**, Volume 17 Issue 3
Full text available: pdf(876.64 KB) Additional Information: full citation, abstract, references, citings, index terms

In this paper we compare the simulated performance of a family of multiprocessor architectures b on a global shared memory. The processors are connected to the memory through caches that sn one or more shared buses in crossbar arrangement. We have simulated a number of configuration order to assess the relative performance of multiple versus wide bus machines, with varying amou prefetch. Four programs, with widely differing characteristics, were run on each confi ...

36 Managing pages in shared virtual memory systems: getting the compiler into the game
Elana D. Granston, Harry A. G. Wijshoff
August 1993 **Proceedings of the 7th international conference on Supercomputing**
Full text available: pdf(1.20 MB) Additional Information: full citation, abstract, references, citings, index terms

In large-scale multiprocessors, whether loosely or tightly coupled, some memory is cheaper to acc than other memory. Because direct management of memory on these machines is quite burdenso the programmer, much research effort has been directed toward providing a shared virtual memo (SVM) interface. Clearly, the success of this endeavor depends heavily on the efficiency of page management strategies. To date, this has been primarily the responsibility of the operating systen s ...

37
A decentralized communication efficient distributed shared memory

Legond L. Burge, Mitchell L. Neilsen
February 1996 **Proceedings of the 1996 ACM symposium on Applied Computing**
Full text available: pdf(717.39 KB) Additional Information: full citation, references, index terms

Keywords: database, distributed algorithm, distributed shared memory, memory coherence, seq consistency

38 Hive: fault containment for shared-memory multiprocessors
J. Chapin, M. Rosenblum, S. Devine, T. Lahiri, D. Teodosiu, A. Gupta
December 1995 **ACM SIGOPS Operating Systems Review , Proceedings of the fifteenth ACM symposium on Operating systems principles**, Volume 29 Issue 5
Full text available: pdf(1.90 MB) Additional Information: full citation, references, citings, index terms

39 Adaptive software cache management for distributed shared memory architectures
John K. Bennett, John B. Carter, Willy Zwaenepoel
May 1990 **ACM SIGARCH Computer Architecture News , Proceedings of the 17th annual international symposium on Computer Architecture**, Volume 18 Issue 3
Full text available: pdf(1.10 MB) Additional Information: full citation, abstract, references, citings, index terms

An adaptive cache coherence mechanism exploits semantic information about the expected or obs access behavior of particular data objects. We contend that, in distributed shared memory system adaptive cache coherence mechanisms will outperform static cache coherence mechanisms. We h examined the sharing and synchronization behavior of a variety of shared memory parallel progra We have found that the access patterns of a large percentage of shared data objects fa ...

40 Mirage: a coherent distributed shared memory design
B. Fleisch, G. Popek
November 1989 **ACM SIGOPS Operating Systems Review , Proceedings of the twelfth ACM symposium on Operating systems principles**, Volume 23 Issue 5
Full text available: pdf(1.63 MB) Additional Information: full citation, abstract, references, citings, index terms

Shared memory is an effective and efficient paradigm for interprocess communication. We are concerned with software that makes use of shared memory in a single site system and its extensio multimachine environment. Here we describe the design of a distributed shared memory (DSM) sy called Mirage developed at UCLA. Mirage provides a form of network transparency to make netwo boundaries invisible for shared memory and is upward compatible with an existing interfac ...

Results 21 - 40 of 83 Result page: previous 1 **2** 3 4 5 next

Subscribe (Full Service) Register (Limited Service, Free) Login

Search: ● The ACM Digital Library ○ The Guide

+ownership +share +cache +memory +coherency +state +m SEARCH

THE ACM DIGITAL LIBRARY

Feedback Report a problem Satisfaction s

Terms used
ownership share cache memory coherency state maintaining multiple processor permant persisting maint

Sort results by relevance
Display results expanded form

Save results to a Binder
Search Tips
□ Open results in a new window

Try an Advanced Search
Try this search in The ACM Gui

Results 41 - 60 of 83 Result page: previous 1 2 **3** 4 5 next

Relevance scale

41 Transactional lock-free execution of lock-based programs
Ravi Rajwar, James R. Goodman
October 2002 **Proceedings of the 10th international conference on Architectural support for programming languages and operating systems**, Volume 36 , 30 , 37 Issue 5 , 5 , 10
Full text available: pdf(1.61 MB) Additional Information: full citation, abstract, references, citings

This paper is motivated by the difficulty in writing correct high-performance programs. Writing sha memory multi-threaded programs imposes a complex trade-off between programming ease and performance, largely due to subtleties in coordinating access to shared data. To ensure correctnes programmers often rely on conservative locking at the expense of performance. The resulting serialization of threads is a performance bottleneck. Locks also interact poorly with thread schedul and faults, r ...

42 Implementation and performance of Munin
John B. Carter, John K. Bennett, Willy Zwaenepoel
September 1991 **ACM SIGOPS Operating Systems Review , Proceedings of the thirteenth ACM symposium on Operating systems principles**, Volume 25 Issue 5
Full text available: pdf(1.46 MB) Additional Information: full citation, abstract, references, citings, index terms

Munin is a distributed shared memory (DSM) system that allows shared memory parallel program executed efficiently on distributed memory multiprocessors. Munin is unique among existing DSM systems in its use of *multiple consistency protocols* and in its use of *release consistency*. In Munin shared program variables are annotated with their expected access pattern, and these annotation then used by the runtime system to choose a consistency protocol best suited to that acc ...

43 Multithreading and value prediction: Speculative lock elision: enabling highly concurrent multithreaded execution
Ravi Rajwar, James R. Goodman
December 2001 **Proceedings of the 34th annual ACM/IEEE international symposium on Microarchitecture**
Full text available: pdf(1.37 MB) Publisher Site Additional Information: full citation, abstract, references, citings

Serialization of threads due to critical sections is a fundamental bottleneck to achieving high performance in multithreaded programs. Dynamically, such serialization may be unnecessary beca these critical sections could have safely executed concurrently without locks. Current processors c fully exploit such parallelism because they do not have mechanisms to dynamically detect such fal inter-thread dependences.We propose *Speculative Lock Elision (SLE),* a novel micro-architectura .

44 Combined performance gains of simple cache protocol extensions
F. Dahlgren, M. Dubois, P. Stenström
April 1994 **ACM SIGARCH Computer Architecture News , Proceedings of the 21ST annual international symposium on Computer architecture**, Volume 22 Issue 2
Full text available: pdf(1.22 MB) Additional Information: full citation, abstract, references, citings, index terms

We consider three simple extensions to directory-based cache coherence protocols in shared-mem multiprocessors. These extensions are aimed at reducing the penalties associated with memory ac and include a hardware prefetching scheme, a migratory sharing optimization, and a competitive-ι mechanism. Since they target different components of the read and write penalties, they can be combined effectively.Detailed architectural simulations using five benchmarks show substantial combined ...

45 Performance analysis of multiprocessor cache consistency protocols using generalized time Petri nets
Mary K. Vernon, Mark A. Holliday
May 1986 **ACM SIGMETRICS Performance Evaluation Review , Proceedings of the 1986 ACI SIGMETRICS joint international conference on Computer performance modellin measurement and evaluation**, Volume 14 Issue 1
Full text available: pdf(1.15 MB) Additional Information: full citation, abstract, references, citings, index terms

We use an exact analytical technique, based on Generalized Timed Petri Nets (GTPNs), to study th performance of shared bus cache consistency protocols for multiprocessors. We develop a general framework within which the key characteristics of the Write-Once protocol and four enhancements have been combined in various ways in the literature can be identified and evaluated. We then quantitatively assess the performance gains for each of the four enhancements. We conside ...

46 Implementing global memory management in a workstation cluster
M. J. Feeley, W. E. Morgan, E. P. Pighin, A. R. Karlin, H. M. Levy, C. A. Thekkath
December 1995 **ACM SIGOPS Operating Systems Review , Proceedings of the fifteenth ACM symposium on Operating systems principles**, Volume 29 Issue 5
Full text available: pdf(1.52 MB) Additional Information: full citation, references, citings, index terms

47 Hardware prediction for data coherency of scientific codes on DSM
J. T. Acquaviva, W. Jalby
November 2000 **Proceedings of the 2000 ACM/IEEE conference on Supercomputing (CDROM]**
Full text available: pdf(142.06 KB) Publisher Site Additional Information: full citation, abstract, references, index terms

This paper proposes a hardware mechanism for reducing coherency overhead occurring in scientifi computations within DSM systems. A first phase aims at detecting, in the address space regular pa (called streams) of coherency events (such as requests for exclusive, shared or invalidation). Once stream is detected at a loop level, regularity of data access can be exploited at the loop level (spa locality) but also between loops (temporal locality). We present a hardwa ...

48 Performance evaluation of memory consistency models for shared-memory multiprocessors
Kourosh Gharachorloo, Anoop Gupta, John Hennessy
April 1991 **Proceedings of the fourth international conference on Architectural support for programming languages and operating systems**, Volume 19 , 25 , 26 Issue 2 , Special Issue
Full text available: pdf(1.71 MB) Additional Information: full citation, references, citings, index terms

49 Using destination-set prediction to improve the latency/bandwidth tradeoff in shared-memor multiprocessors
Milo M. K. Martin, Pacia J. Harper, Daniel J. Sorin, Mark D. Hill, David A. Wood
May 2003 **ACM SIGARCH Computer Architecture News , Proceedings of the 30th annual international symposium on Computer architecture**, Volume 31 Issue 2
Full text available: pdf(220.76 KB) Additional Information: full citation, abstract, references, citings

Destination-set prediction can improve the latency/bandwidth tradeoff in shared-memory multiprocessors. The destination set is the collection of processors that receive a particular cohere request. Snooping protocols send requests to the maximal destination set (i.e., all processors), re latency for cache-to-cache misses at the expense of increased traffic. Directory protocols send req to the minimal destination set, reducing bandwidth at the expense of an indirection through the d

50 Boosting the performance of hybrid snooping cache protocols
Fredrik Dahlgren
May 1995 **ACM SIGARCH Computer Architecture News , Proceedings of the 22nd annual international symposium on Computer architecture**, Volume 23 Issue 2
Full text available: pdf(1.23 MB) Additional Information: full citation, abstract, references, citings, index terms

Previous studies of bus-based shared-memory multiprocessors have shown hybrid write-invalidate/write-update snooping protocols to be incapable of providing consistent performance improvements over write-invalidate protocols. In this paper, we analyze the deficiencies of hybrid snooping protocols under release consistency, and show how these deficiencies can be dramaticall reduced by using write caches and read snarfing.Our performance evaluation is based on program driven simulation and a set o ...

51 Optimizing software cache-coherent cluster architectures
Xiaohan Qin, Jean-Loup Baer
November 1998 **Proceedings of the 1998 ACM/IEEE conference on Supercomputing (CDROM)**
Full text available: html(53.87 KB) Additional Information: full citation, abstract, references

Software cache-coherent systems using programmable protocol processors provide a flexible infrastructure to expand the systems in size and function. However this flexibility comes at a cost performance. First, the software implementation of protocols is inherently slower than a hardware implementation. Second, when multiple processors share a protocol processor, contention may re a substantial increase in memory latency.In this paper, we study how the overhead of a software scheme can ...

Keywords: communication primitives, performance evaluation, software-controlled cache coherei

52 Options for dynamic address translation in COMAs
Xiaogang Qiu, Michel Dubois
April 1998 **ACM SIGARCH Computer Architecture News , Proceedings of the 25th annual international symposium on Computer architecture**, Volume 26 Issue 3
Full text available: pdf(1.37 MB) Publisher Site Additional Information: full citation, abstract, references, citings, index terms

In modern processors, the dynamic translation of virtual addresses to support virtual memory is d before or in parallel with the first-level cache access. As processor technology improves at a rapid and the working sets of new applications grow insatiably the latency and bandwidth demands on t (Translation Lookaside Buffer) are getting more and more difficult to meet. The situation is worse multiprocessor systems, which run larger applications and are plagued by the TLB consiste ...

53 Performance of cache coherence in stackable filing
J. Heidemann, G. Popek

December 1995 **ACM SIGOPS Operating Systems Review , Proceedings of the fifteenth ACM symposium on Operating systems principles**, Volume 29 Issue 5

Full text available: pdf(2.00 MB) Additional Information: full citation, references, index terms

54 Delayed consistency and its effects on the miss rate of parallel programs

Michel Dubois, Jin Chin Wang, Luiz A. Barroso, Kangwoo Lee, Yung-Syau Chen

August 1991 **Proceedings of the 1991 ACM/IEEE conference on Supercomputing**

Full text available: pdf(1.01 MB) Additional Information: full citation, references, citings, index terms

55 Cache coherence in systems with parallel communication channels & many processors

John C. Willis, Arthur C. Sanderson, Charles R. Hill

November 1990 **Proceedings of the 1990 ACM/IEEE conference on Supercomputing**

Full text available: pdf(868.59 KB) Additional Information: full citation, abstract, references

This paper describes and analyzes two algorithms for maintaining cache coherence in multiprocess systems with parallel communication channels and many processors. A distributed link-list relates cache frames representing the same main memory block. Messages traverse the list to maintain li integrity, exclusive ownership, and consistent values. Memory access semantics are equivalent to shared memory system without caches. Reference latency, efficiency of memory use, and hardwa complex ...

56 An economical solution to the cache coherence problem

James Archibald, Jean Loup Baer

January 1984 **ACM SIGARCH Computer Architecture News , Proceedings of the 11th annual international symposium on Computer architecture**, Volume 12 Issue 3

Full text available: pdf(728.73 KB) Additional Information: full citation, abstract, references, citings, index terms

In this paper we review and qualitatively evaluate schemes to maintain cache coherence in tightly coupled multiprocessor systems. This leads us to propose a more economical (hardware-wise), expandable and modular variation of the "global directory" approach. Protocols for this solution ar described. Performance evaluation studies indicate the limits (number of processors, level of shari within which this approach is viable.

57 The detection and elimination of useless misses in multiprocessors

Michel Dubois, Jonas Skeppstedt, Livio Ricciulli, Krishnan Ramamurthy, Per Stenström

May 1993 **ACM SIGARCH Computer Architecture News , Proceedings of the 20th annual international symposium on Computer architecture**, Volume 21 Issue 2

Full text available: pdf(1.03 MB) Additional Information: full citation, abstract, references, citings, index terms

In this paper we introduce a new classification of misses in shared-memory multiprocessors based interprocessor communication. We identify the set of essential misses, i.e., the smallest set of mis necessary for correct execution. Essential misses include cold misses and true sharing misses. All misses are useless misses and can be ignored without affecting the correctness of program execut Based on the new classification we compare the effectiveness of five different protoc ...

58 Parallel architectures: Inferential queueing and speculative push for reducing critical communication latencies

Ravi Rajwar, Alain Kägi, James R. Goodman

June 2003 **Proceedings of the 17th annual international conference on Supercomputing**

Full text available: pdf(568.93 KB) Additional Information: full citation, abstract, references, index terms

Communication latencies within critical sections constitute a major bottleneck in some classes of

emerging parallel workloads. In this paper, we argue for the use of Inferentially Queued Locks (IQ
[31], not just for efficient synchronization but also for reducing communication latencies, and we
propose a novel mechanism, Speculative Push (SP), aimed at reducing these communication laten
With IQLs, the processor infers the existence, and limits, of a critical section from the use of synch

Keywords: data forwarding, inferential queueing, synchronization

59 Using "test model-checking" to verify the Runway-PA8000 memory model
Rajnish Ghughal, Abdel Mokkedem, Ratan Nalumasu, Ganesh Gopalakrishnan
June 1998 **Proceedings of the tenth annual ACM symposium on Parallel algorithms and architectures**
Full text available: pdf(1.14 MB) Additional Information: full citation, references, citings, index terms

60 The impact of architectural trends on operating system performance
M. Rosenblum, E. Bugnion, S. A. Herrod, E. Witchel, A. Gupta
December 1995 **ACM SIGOPS Operating Systems Review , Proceedings of the fifteenth ACM symposium on Operating systems principles**, Volume 29 Issue 5
Full text available: pdf(2.03 MB) Additional Information: full citation, references, citings, index terms

Results 41 - 60 of 83 Result page: previous 1 2 **3** 4 5 next

PORTAL

US Patent & Trademark Office

Subscribe (Full Service) Register (Limited Service, Free) Login

Search: ● The ACM Digital Library ○ The Guide

+ownership +share +cache +memory +coherency +state +m SEARCH

THE ACM DIGITAL LIBRARY

Feedback Report a problem Satisfaction s

Terms used
ownership share cache memory coherency state maintaining multiple processor permant persisting maint

Sort results by relevance

Display results expanded form

Save results to a Binder

Search Tips

□ Open results in a new window

Try an Advanced Search
Try this search in The ACM Gui

Results 61 - 80 of 83 Result page: previous 1 2 3 **4** 5 next

Relevance scale

61 An accurate and efficient performance analysis technique for multiprocessor snooping cache consistency protocols
M. K. Vernon, E. D. Lazowska, J. Zahorjan
May 1988 **ACM SIGARCH Computer Architecture News , Proceedings of the 15th Annual International Symposium on Computer architecture**, Volume 16 Issue 2
Full text available: pdf(999.88 KB) Additional Information: full citation, abstract, references, citings, index terms

A number of dynamic cache consistency protocols have been developed for multiprocessors having shared bus interconnect between processors and shared memory. The relative performance of the protocols has been studied extensively using simulation and detailed analytical models based on M chain techniques. Both of these approaches use relatively detailed models, which capture cache ar interference rather precisely, but which are highly expensive to evaluate. In this paper, we inv ...

62 Multiprocessor cache analysis using ATUM
R. L. Sites, A. Agarwal
May 1988 **ACM SIGARCH Computer Architecture News , Proceedings of the 15th Annual International Symposium on Computer architecture**, Volume 16 Issue 2
Full text available: pdf(1.38 MB) Additional Information: full citation, abstract, references, citings, index terms

The design of high-performance multiprocessor systems necessitates a careful analysis of the mer system performance of parallel programs. Lacking multiprocessor address traces, previous multiprocessor performance studies using analytical models had to make an inordinate number of assumptions about the underlying memory reference patterns. We previously developed a scheme ATUM - Address Tracing Using Microcode - to get reliable operating system and multiprogramming traces on single ...

63 Supporting reference and dirty bits in SPUR's virtual address cache
D. A. Wood, R. H. Katz
April 1989 **ACM SIGARCH Computer Architecture News , Proceedings of the 16th annual international symposium on Computer architecture**, Volume 17 Issue 3
Full text available: pdf(1.12 MB) Additional Information: full citation, abstract, references, citings, index terms

Virtual address caches can provide faster access times than physical address caches, because tran is only required on cache misses. However, because we don't check the translation information on cache access, maintaining reference and dirty bits is more difficult. In this paper we examine the t offs in supporting reference and dirty bits in a virtual address cache. We use measurements from uniprocessor SPUR prototype to evaluate different alternatives. The prototype's buil ...

64 Mapping irregular applications to DIVA, a PIM-based data-intensive architecture
Mary Hall, Peter Kogge, Jeff Koller, Pedro Diniz, Jacqueline Chame, Jeff Draper, Jeff LaCoss, John Gr
Jay Brockman, Apoorv Srivastava, William Athas, Vincent Freeh, Jaewook Shin, Joonseok Park
January 1999 **Proceedings of the 1999 ACM/IEEE conference on Supercomputing (CDROM)**
Full text available: pdf(111.41 KB) Additional Information: full citation, references, citings, index terms

65 Shared memory computing on SP2: JIAJIA approach
M. Rasit Eskicioglu, T. Anthony Marsland
November 1998 **Proceedings of the 1998 conference of the Centre for Advanced Studies on Collaborative research**
Full text available: pdf(99.27 KB) Additional Information: full citation, abstract, references, index terms

Distributed shared memory (DSM) is a useful abstraction not only for deploying networks of workstations as a parallel multicomputer but also for increasing the usability of non-uniform mem access multicomputers. It provides an alternative programming model for distributed memory computers. In this paper, we present empirical evaluation of JIAJIA, a software DSM system, on a SP2 cluster. We also discuss the performance of a suite of six widely different applications running this sof ...

66 Comparative evaluation of latency reducing and tolerating techniques
Anoop Gupta, John Hennessy, Kourosh Gharachorloo, Todd Mowry, Wolf-Dietrich Weber
April 1991 **ACM SIGARCH Computer Architecture News , Proceedings of the 18th annual international symposium on Computer architecture**, Volume 19 Issue 3
Full text available: pdf(1.36 MB) Additional Information: full citation, references, citings, index terms

67 The Amber system: parallel programming on a network of multiprocessors
J. Chase, F. Amador, E. Lazowska, H. Levy, R. Littlefield
November 1989 **ACM SIGOPS Operating Systems Review , Proceedings of the twelfth ACM symposium on Operating systems principles**, Volume 23 Issue 5
Full text available: pdf(1.53 MB) Additional Information: full citation, abstract, references, citings, index terms

This paper describes a programming system called Amber that permits a single application progra use a homogeneous network of computers in a uniform way, making the network appear to the application as an integrated multiprocessor. Amber is specifically designed for high performance ir case where each node in the network is a shared-memory multiprocessor. Amber shows that supp loosely-coupled multiprocessing can be efficiently realized using an obje ...

68 A cost-comparison approach for adaptive distributed shared memory
Jai-Hoon Kim, Nitin H. Vaidya
January 1996 **Proceedings of the 10th international conference on Supercomputing**
Full text available: pdf(976.97 KB) Additional Information: full citation, references, citings, index terms

69 Tolerating latency in multiprocessors through compiler-inserted prefetching
Todd C. Mowry
February 1998 **ACM Transactions on Computer Systems (TOCS)**, Volume 16 Issue 1
Full text available: pdf(410.70 KB) Additional Information: full citation, abstract, references, citings, index terms,

The large latency of memory accesses in large-scale shared-memory multiprocessors is a key obst achieving high processor utilization. Software-controlled prefetching is a technique for tolerating memory latency by explicitly executing instructions to move data close to the processor before the are actually needed. To minimize the burden on the programmer, compiler support is needed to automatically insert prefetch instructions into the code. A key challenge when ...

Keywords: compiler optimization, prefetching

70 Diffracting trees
Nir Shavit, Asaph Zemach
November 1996 **ACM Transactions on Computer Systems (TOCS)**, Volume 14 Issue 4
Full text available: pdf(729.57 KB) Additional Information: full citation, abstract, references, citings, index terms

Shared counters are among the most basic coordination structures in multiprocessor conputation, applications ranging from barrier synchronization to concurrent-data-structure design. This article introduces diffracting trees, novel data structures for share counting and load balancing in a distributed/parallel environment. Empirical evidence, collected on a simulated distributed shared-memory machine and several simulated message-passing architectures, shows that diffracting tre scal ...

Keywords: contention, counting networks, index distribution, lock free, wait free

71 A low-overhead coherence solution for multiprocessors with private cache memories
Mark S. Papamarcos, Janak H. Patel
January 1984 **ACM SIGARCH Computer Architecture News , Proceedings of the 11th annual international symposium on Computer architecture**, Volume 12 Issue 3
Full text available: pdf(590.93 KB) Additional Information: full citation, abstract, references, citings, index terms

This paper presents a cache coherence solution for multiprocessors organized around a single time shared bus. The solution aims at reducing bus traffic and hence bus wait time. This in turn increas overall processor utilization. Unlike most traditional high-performance coherence solutions, this so does not use any global tables. Furthermore, this coherence scheme is modular and easily extensi requiring no modification of cache modules to add more processors to a system. The ...

72 Object race detection
Christoph von Praun, Thomas R. Gross
October 2001 **ACM SIGPLAN Notices , Proceedings of the 16th ACM SIGPLAN conference on O oriented programming, systems, languages, and applications**, Volume 36 Issue 11
Full text available: pdf(261.72 KB) Additional Information: full citation, abstract, references, citings, index terms

We present an on-the-fly mechanism that detects access conflicts in executions of multi-threaded programs. Access conflicts are a conservative approximation of data races. The checker tracks acc information at the level of objects (*object races*) rather than at the level of individual variables. Th viewpoint allows the checker to exploit specific properties of object-oriented programs for optimiz by restricting dynamic checks to those objects that are identified by escape an ...

73 A low-overhead coherence solution for multiprocessors with private cache memories
Mark S. Papamarcos, Janak H. Patel
August 1998 **25 years of the international symposia on Computer architecture (selected pap**
Full text available: pdf(707.80 KB) Additional Information: full citation, references, index terms

74 A single cached copy data coherence scheme for multiprocessor systems
A. Mendelson, D. K. Pradhan, A. D. Singh
December 1989 **ACM SIGARCH Computer Architecture News**, Volume 17 Issue 6
Full text available: Additional Information:

pdf(667.24 KB) full citation, abstract, index terms

We present and evaluate a snoopy cache memory protocol, the Single Cache Copy Data Coherenc (SCCDC), for multiprocessors that allows only a single cache to hold a given share-d data at any ti The simulations presented here indicate that despite its simplicity, the scheme has the potential fc performance comparable with more complex snoopy cache schemes. We have also shown in relate work [8] that the existence of only a single copy of data in cache allows efficient access control to

75 The architecture of a Linda coprocessor
V. Krishnaswamy, S. Ahuja, N. Carriero, D. Gelernter
May 1988 **ACM SIGARCH Computer Architecture News , Proceedings of the 15th Annual International Symposium on Computer architecture**, Volume 16 Issue 2
Full text available: pdf(1.09 MB) Additional Information: full citation, abstract, references, citings, index terms

We describe the architecture of a coprocessor that supports the communication primitives of the L parallel programming environment in hardware. The coprocessor is a critical element in the archit of the Linda Machine, an MIMD parallel processing system that is designed top down from the specifications of Linda. Communication in Linda programs takes place through a logically shared associative memory mechanism called tuple space. The Linda Machine, however, has no physically shared ...

76 The Starfire SMP interconnect
Alan Charlesworth, Nicholas Aneshansley, Mark Haakmeester, Dan Drogichen, Gary Gilbert, Ricki Wil Andrew Phelps
November 1997 **Proceedings of the 1997 ACM/IEEE conference on Supercomputing (CDROM)**
Full text available: pdf(273.52 KB) Additional Information: full citation, abstract, references, citings

The Starfire interconnect extends the envelope of Unix symmetric multiprocessor (SMP) systems i several dimensions. **Interconnect:** an active centerplane with four address routers and a 16x16 c crossbar provides 64 UltraSPARC processors with uniform memory access at a bandwidth of 10,66 MBps. **Flexibility:** Starfire can be dynamically reconfigured into multiple hardware-protected oper system domains. **Robustness:** Failing boards can be hot swapped without interrupting sy ...

Keywords: SMP, UMA, bandwidth, domains, interconnect, latency, partitions

77 On the validity of trace-driven simulation for multiprocessors
Eric J. Koldinger, Susan J. Eggers, Henry M. Levy
April 1991 **ACM SIGARCH Computer Architecture News , Proceedings of the 18th annual international symposium on Computer architecture**, Volume 19 Issue 3
Full text available: pdf(840.99 KB) Additional Information: full citation, references, citings, index terms

78 The K2 distributed memory parallel processor: architecture, compiler, and operating system
M. Annaratone, M. Fillo, M. Halbherr, R. Rühl, P. Steiner, M. Viredaz
August 1991 **Proceedings of the 1991 ACM/IEEE conference on Supercomputing**
Full text available: pdf(1.13 MB) Additional Information: full citation, references, citings, index terms

79 The evolution of Coda
M. Satyanarayanan
May 2002 **ACM Transactions on Computer Systems (TOCS)**, Volume 20 Issue 2
Full text available: pdf(441.35 KB) Additional Information: full citation, abstract, references, citings, index terms

Failure-resilient, scalable, and secure read-write access to shared information by mobile and static over wireless and wired networks is a fundamental computing challenge. In this article, we descrit the Coda file system has evolved to meet this challenge through the development of mechanisms server replication, disconnected operation, adaptive use of weak connectivity, isolation-only transactions, translucent caching, and opportunistic exploitation of hardware surrogates. For eac .

Keywords: Adaptation, Linux, UNIX, Windows, caching, conflict resolution, continuous data acces data staging, disaster recovery, disconnected operation, failure, high availability, hoarding, interm networks, isolation-only transactions, low-bandwidth networks, mobile computing, optimistic replic control, server replication, translucent cache management, weakly connected operation

80 Are crossbars really dead?: the case for optical multiprocessor interconnect systems
Andreas G. Nowatzyk, Paul R. Prucnal
May 1995 **ACM SIGARCH Computer Architecture News , Proceedings of the 22nd annual international symposium on Computer architecture**, Volume 23 Issue 2
Full text available: pdf(1.16 MB) Additional Information: full citation, abstract, references, citings, index terms

Crossbar switches are rarely considered for large, scalable multiprocessor interconnect systems be they require $O(n^2)$ switching elements, are difficult to control efficiently and are hard to implement their size becomes too large to fit on one integrated circuit. However these problems are technolo dependent and a recent innovation in fiber optic devices has led to a new implementation of crossl switches that does not share these problems while retaining the full advanta ...

Results 61 - 80 of 83 Result page: previous 1 2 3 **4** 5 next